AD659484

AN AXIOMATIC THEORY OF GENERAL SYSTEMS*

T. G. Windeknecht

Case Western Reserve University

OCT 1 ∂ 1967

Abstract

The paper presents a formalization of the principal concepts of the block diagram approach in systems theory using set theory. The development is axiomatic. Starting from Mesarovic's notion of a general system as an n-ary relation, (i) the concept of time is introduced (ii) multi-variable input-output systems are formalized and (iii) the evolution of such systems in time is studied both with and without the property of non-anticipation. It is demonstrated in the latter case that the concept of state naturally arises.

Introduction

In the companion paper, Professor Mesarović indicates the existence of a gulf between the block diagram analysis and the analysis of detailed mathematical models for complex systems. Moreover, he suggests that general systems theory can properly play a role in bridging this gap "by preserving the simplicity of the block diagram while introducing the precision of mathematics." As to achieving this goal, he provides us with several significant clues. Most important among these are (i) that set theory is an appropriate mathematical vehicle (augmented perhaps with some elementary concepts from abstract algebra) and (ii) that fundamentally a system is a relation between input set and output set. To these, we add (iii) essentially all systems of interest in engineering exist and operate within some reference-frame of time.

Now, if the gulf is genuinely to be bridged, it is apparent that the principal concepts of the block diagram point-of-view must be incorporated into general systems theory. In fact, this must be carried out very carefully so that one can move easily from the block diagram to the general systems theory set-up. Otherwise, the general systems theory set-up certainly cannot serve as a bridge in the design process. Thus, we arrive at the position that a fundamental task to be undertaken in general systems theory is the formalization of block diagram concepts using set theory.

Our purpose in this article is to present such a formalization and to discuss a result within the framework of this formalization which we have recently reported [1]. Here; we shall attempt to make our presentation understandable to some not thoroughly versed in axiomatic set theory. Moreover, we shall be very directly concerned with defending the position that our formalization of the block diagram concepts is a reasonable one

intuitively. Hence, we shall devote considerable attention to the
interpretation of various mathematical objects within our formalism.

Another remark about our overall approach is in order. The block
diagram concepts we seek to formalize (such as system, input, output,
state, etc.) are very general concepts to begin with. Their meaning
in engineering, while not completely precise, is reasonably well-
understood. It is possible, therefore, to proceed in an axiomatic
fashion with the development, i.e. to go from the general to the more
specific. To do so is both attractive and pedagogically dangerous.
It is attractive because there is associated with an axiomatic theory
a powerful mechanism for properly identifying basic concepts. It is
dangerous because things become abstract. There is, we feel, sufficient
uncertainty as to which concepts in systems theory are basic that an
axiomatic approach is justified despite the abstractness. Moreover,
as Professor Mesarović has pointed out, the use of such a general
systems theory development in the engineering design process appears
to occur in a flow which goes from the less specific (block diagram)
model to the more specific (detailed mathematical) model. Thus, the
theory may be of greatest utility if it is constructed axiomatically.

Notation

In order to render the presentation concise, it is necessary to
assume some basic ideas from set theory; namely, the concepts of sets,
functions, and relations and the elementary facts about these.[1] Moreover,
we shall use the standard notations regarding sets, functions, and

---

[1] All of the essential material is presented in chapters 1-3 of
Suppes [2].

relations without introduction.

Some particular notations we adopt are the following: We denote unit sets without brackets, ordered sets (i.e. n-tuples) with parentheses, and ordinary sets with curly brackets. If S is a 2-ary relation, i.e. a set of ordered pairs, then the sets

$$\mathcal{D}S = \{x \mid (\exists y): xSy\}$$

and
$$\mathcal{R}S = \{y \mid (\exists x): xSy\}$$

are, respectively, the <u>domain</u> and the <u>range</u> of S. Here $xSy \Leftrightarrow (x,y) \in S$. In general, then, $S \subseteq \mathcal{D}S \times \mathcal{R}S$ (where $\mathcal{D}S \times \mathcal{R}S = \{(x,y) \mid x \in \mathcal{D}S \ \& \ y \in \mathcal{R}S\}$) and, if $S \subseteq X \times Y$, then $\mathcal{D}S \subseteq X$ and $\mathcal{R}S \subseteq Y$.

The <u>composition</u> of two 2-ary relations S and S' is the set

$$(S' \circ S) = \{(x,z) \mid (\exists y): xSy \ \& \ yS'z\}$$

which is itself a 2-ary relation. From the definition, $\mathcal{D}(S' \circ S) \subseteq \mathcal{D}S$ and $\mathcal{R}(S' \circ S) \subseteq \mathcal{R}S'$ and, in particular, the condition $\mathcal{R}S \subseteq \mathcal{D}S'$ implies $\mathcal{D}(S' \circ S) = \mathcal{D}S$. If S, S', and S'' are 2-ary relations, then we see

$$(S'' \circ S') \circ S = S'' \circ (S' \circ S)$$

In the case that $S: X \rightarrow Y$ and $S': Z \rightarrow W$ are functions, $(S' \circ S)$ is a function and

$$\mathcal{D}(S' \circ S) = \{x \mid x \in X \ \& \ S(x) \in Z\}$$

Moreover, for all $x \in \mathcal{D}(S' \circ S)$,

$$(S' \circ S)(x) = S'(S(x))$$

We shall reserve the word <u>transformation</u> for a function which maps a set into itself. If f and g are transformations on the set Y, then $\mathcal{D}(f \circ g) = Y$. Indeed, if $f: Y \rightarrow Y$ and $g: Y \rightarrow Y$, then $(f \circ g): Y \rightarrow Y$. Thus, the composition of two transformations on a set Y is a transformation

on Y. Even more, the composition of two (onto; 1:1; 1:1 onto) transformations is (onto; 1:1; 1:1 onto).

Finally, if X and Y are sets, we write Y^X to denote the class of all functions which map X into Y, i.e.

$$Y^X = \{f \mid f:X \to Y\}$$

For example, if Y is a set, then Y^Y is the class of all transformations on Y.

Time Sets

Recall our earlier stated position (iii) above; namely, that "essentially all systems of interest in engineering exist and operate within some reference-frame of time." This assumption will have a major effect on our formalization of the systems concepts. In particular, it will be through the explicit introduction of a mathematical representation for time that we put structure into the concept of a system beyond that asserted by Professor Mesarović in his article. Our approach will be to introduce a special kind of a set to represent time.

Now, to represent time in engineering models for systems, one ordinarily uses one of the following four sets (i) the set R of real numbers (ii) the set R+ of positive real numbers (iii) the set I of integers or (iv) the set I+ of positive integers. For our purposes, none of these sets is actually suitable. In fact, to choose any one of these is to lose an important degree of generality. That is, to choose R or R+ can at best lead us to the so-called "continuous-time" systems and with I or I+ we shall have formalized only the "discrete-time" systems. Either choice would severely undermine our objectives.

Instead of choosing one of the usual sets, we can take an abstract set which is postulated to have the essential features that R, R+, I,

and I+ share and which are important in representing time. This will allow us to discuss "continuous-time" systems and "discrete-time" systems simultaneously, i.e. by specialization, we can explicitly exhibit either case. The essential features of a time set apparently include (i) a simple ordering (the inequality relation $<$ in the case of R, R+, I, and I+) and (ii) an algebraic operation (ordinary addition in the case of R, R+, I, and I+). A concept conveniently satisfying these require- ments is the concept in abstract algebra of an ordered group. Since many of our readers may be unfamiliar with this concept, we repeat the definition here and develop some tools to be used in our later work.[2]

An ordered pair of sets $(T, +)$ is a <u>group</u> if $(+)$ is an operation on T (i.e. if $+: T \times T \to T$) such that (i) for all $t, t', t'' \in T$, $(t + t') + t'' = t + (t' + t'')$ (ii) there exists some $0 \in T$ (called an <u>identity</u>) such that for all $t \in T$, $(t + 0) = (0 + t) = t$ and (iii) for every $t \in T$ there exists some $t^{-1} \in T$ such that $(t + t^{-1}) = (t^{-1} + t) = 0$. In general, it can be shown that the identity element 0 is unique in T and each element t^{-1} is unique given t. More- over, if $(T, +)$ is a group, then for all $t, t' \in T$, the identities $(t^{-1})^{-1} = t$ and $(t + t')^{-1} = t'^{-1} + t^{-1}$ hold.

For example, if G is the set of all 1:1 onto transformations on a set A, then (G, o) is a group where (o) is composition of transformations. In particular, the identity function on A (i.e. the set $c_0 = \{(a, a) \mid a \in A\}$) serves as identity in G and, if $g \in G$, then

$$g^{-1} = \{(g(a), a) \mid a \in A\}$$

which is an element of G since g is 1:1 onto A. More familiar examples of groups are $(R, +)$ and $(I, +)$ where R is the set of real numbers, I is

[2] We follow Jacobson [3].

the set of integers, and (+) is ordinary addition of real numbers.

If $(T,+)$ and $(T',+')$ are groups, then a map $h:T \to T'$ (into) is a
homomorphism if for all $t,t' \in T$, $h(t + t') = h(t) +' h(t')$. An iso-
morphism is a homomorphism which is 1:1 on T onto T'. $(T,+)$ and $(T',+')$
are homomorphic [isomorphic] if and only if there exists a homomorphism
[isomorphism] $h:T \to T'$. Thus, homomorphic and isomorphic groups are
sets which are algebraically similar.

A very basic result from group theory which we shall use extensively
is the notion of the "Cayley representation" of a group. If $(T,+)$ is a
group and $t \in T$, then the t-left translation on T is the transformation
$e_t:T \to T$ such that for all $t' \in T$, $e_t(t') = t + t'$. The Cayley represent-
ation of $(T,+)$ is the ordered pair (C_T,o) where (o) is composition and
$C_T = \{e_t \mid t \in T\}$. It can be proved in general: If $(T,+)$ is a group,
then (C_T,o) is an isomorphic group [4]. In particular, the map which
takes $t \to e_t$ is an isomorphism. It follows for all $t \in T$, that (i)
$e_0(t) = t$ (whence e_0 is the identity function on T) (ii)
$(e_t)^{-1} = e_{t^{-1}}$ and (iii) e_t is 1:1. Also, of course, for all $t,t' \in T$,
$e_{t+t'} = e_t \circ e_{t'}$. C_T is thus a set of 1:1 onto transformations on T.

Our final preliminary is to introduce the concept of a group with
a simple ordering and consider briefly the Cayley representation of such
a group. An ordered group is a group $(T,+)$ together with a subset $U \subseteq T$
such that (i) the set $\{U, 0, U^{-1}\}$ is a partition of T, where

$$U^{-1} = \{t^{-1} \mid t \in U\}$$

and (ii) $+:U \times U \to U$. If $(T,+)$ is a group ordered by U, then the relation
$(<)$ on T such that

$$t < t' <> (t^{-1} + t') \in U, \qquad (t,t' \in T)$$

satisfies (i) $t < t'$ & $t' < t'' \to t < t''$ (ii) $t < t' \lor t = t' \lor t' < t$

and (iii) $t < t' \Leftrightarrow (t'' + t) < (t'' + t')$ for all $t, t', t'' \in T$.

Thus, $(<)$ is what is termed a left invariant, simple ordering of T [2]. Finally, it turns out: $U = \{t \mid 0 < t\}$.

Now, if $(T, +)$ is a group ordered by U, then the Cayley representation $(C_T, \circ)$ is ordered by the set $C_U = \{e_t \mid t \in U\}$. In this case, all of the transformations $e_t \in C_T$ preserve the ordering $(<)$ on T. In fact, since $(<)$ is left invariant, for all $t, t', t'' \in T$ we have

$$t' < t'' \Leftrightarrow (t + t') < (t + t'') \Leftrightarrow e_t(t') < e_t(t'')$$

Thus, C_T is a collection of monotonic (i.e. order-preserving) 1:1 onto transformations on T. Now, if $(\prec)$ denotes the ordering of C_T induced by C_U, i.e.

$$e_t \prec e_{t'} \Leftrightarrow ((e_t)^{-1} \circ e_{t'}) \in C_U$$

then we have

$$t < t' \Leftrightarrow e_t \prec e_{t'} \qquad\qquad (t, t' \in T)$$

In fact,

$$t < t' \Leftrightarrow (t^{-1} + t') \in U \Leftrightarrow e_{t^{-1}+t'} \in C_U \Leftrightarrow (e_{t^{-1}} \circ e_{t'}) \in C_U$$
$$\Leftrightarrow ((e_t)^{-1} \circ e_{t'}) \in C_U \Leftrightarrow e_t \prec e_{t'}$$

Hence, the isomorphism h which takes $t \to e_t$ is also order-preserving. Such an isomorphism is called an order-isomorphism [3].

Of course, two important examples of ordered groups are (i) the additive group of real numbers $(R, +)$ which is ordered by the set of positive reals $R+$, and (ii) the additive group of integers $(I, +)$ which is ordered by the set of positive integers $I+$. In each case, the simple ordering induced is the ordinary strict inequality relation $<$. Thus, we see that the concept of an ordered group encompasses all four of the usual time sets. It is important to note that somewhat different roles are played by $R+$ and $I+$ in the concept of an ordered group than R and I play.

General Time Systems

Henceforth, we can deal straight on with the formalization of block diagram concepts. Our first task is to set down a working definition of a general time system (which the reader will recognize to be a specialization of Professor Mesarovic's definition of a system as a general relation). Having elaborated on this definition somewhat (in particular, having carefully identified in systems theory terminology our interpretation of the various sets associated with a general time system), our approach will be to axiomatically develop the concept of state for our systems. As in [1] , our main point is to show that states and state transitions in systems theory arise naturally in an exceedingly general case. In fact, the only axiom required is the property of non-anticipation. Since one is not far wrong to equate non-anticipation of a system with its "physical realizability", we thus infer that all physically realizable systems have states and functional state transitions and, even more, these state transitions obey a well-known rule (called the "semi-group" property [5]). More explicitly than we were able to accomplish in [1] , we demonstrate that the state concept in systems theory is completely interwoven with the concept of the evolution of an input-output relation (i.e. a system) in time. This would seem once and for all to justify the study of states and state transitions in systems theory as a means of analyzing how a system will appear at various future instants of time.

We proceed to our working definition of a general time system: Henceforth, let $G = (T, +, U, <)$ be an ordered group, i.e. let $(T, +)$ be a group ordered by the subset U and let $(<)$ be the simple ordering of T induced by U. If n is a positive integer and $(A_1, A_2, \ldots, A_n)$ is an n-tuple of sets, then any non-empty relation $S \subseteq (A_1^U \times A_2^U \times \ldots \times A_n^U)$ is an n-ary G-system. Here, as usual,

$$A_i^U = \{s_i \mid s_i : U \to A_i\}$$

and $(\times)$ denotes Cartesian product. If $S \subseteq (A_1^U \times A_2^U \times \ldots \times A_n^U)$ is a G-system, then the sets $A_1, A_2, \ldots, A_n$ are called __spaces__ of S, the set U is the __time set__ of S, and the identity element O of T is the __starting time__ of S. n is the __degree__ of S. A G-system S is __discrete-time__ if $(T,+)$ is order-isomorphic[3] with the additive group $(I,+)$ of integers. It is __continuous-time__ if $(T,+)$ is order-isomorphic with the additive group $(R,+)$ of the real numbers. Evidently, no G-system is both discrete-time and continous-time.

Let S be an n-ary G-system. If $1 \leq i \leq n$, then the __i-th projection__ of S is the set

$$P_i S = \{s_i \mid (\exists s_1) \ldots (\exists s_{i-1})(\exists s_{i+1}) \ldots (\exists s_n): (s_1, \ldots, s_i, \ldots, s_n) \in S\}$$

S is __improper__ if

$$S = P_1 S \times P_2 S \times \ldots \times P_n S$$

Evidently, if S is an n-ary G-system and $1 \leq i \leq n$, then $P_i S$ is a unary G-system. In general,

$$S \subseteq P_1 S \times P_2 S \times \ldots \times P_n S$$

Finally, every unary G-system is improper.

Thus, we propose the n-ary G-system as a formalization of the concept of a general time system. In doing this, we note (above) that n-ary G-systems can be continuous-time or discrete-time if we specialize the ordered group G appropriately. Finally, in identifying the class of improper n-ary G-systems, we distinguish that subclass whose elements are trivial in a relational sense. We remark that the condition

_____

[3] That is, if there exists an isomorphism $h: T \to I$ which preserves ordering.

$$S = P_1 S \times P_2 S \times \ldots \times P_n S$$

(i.e. that S is improper) always bears the interpretation that the sets $P_i S$ are "independent" of each other (with respect to S). This is a useful concept to be able to formalize.

Time Objects

We can by specialization of the concept of a general time system arrive at Professor Mesarovic's concept of the "objects" of systems. In our case, these "objects" are "time objects", i.e. sets of (generalized) time functions. We define:

A set V is a G-object if and only if V is a unary G-system. If V is a G-object and $v \in V$, then v is a G-time function. Evidently, then, a set V is a G-object if and only if V is nonempty and there exists a set A such that $V \subseteq A^U = \{ v \mid v : U \to A \}$. A G-time function is in fact then a function and, in particular, a function with domain U. Now, if S is an n-ary G-system and $1 \leq i \leq n$, then $P_i S$ is a G-object and its elements are G-time functions. Thus, an n-ary G-system is a collection of n-tuples of G-time functions.

Time objects and their elements can be "composite" objects or elements, i.e. they can sometimes be reduced to components: Let $V \subseteq A^U$ be a G-object. If n is a positive integer, then V is n-ary if and only if there exists an n-tuple of sets $(A_1, A_2, \ldots, A_n)$ such that

$$A \subseteq A_1 \times A_2 \times \ldots \times A_n$$

In other words, a G-object is n-ary if its space is a set of n-tuples. n is the dimension of V. If $A \subseteq A_1 \times A_2 \times \ldots \times A_n$ and $1 \leq i \leq n$, then the i-th component map on A is the function $c_i^A : A \to A_i$ such that

$$c_i^A(a_1, a_2, \ldots, a_n) = a_i$$

If V is an n-ary G-object and $v \in V$, then the composition $(c_i^A \circ v)$ is the i-th component of v. Also, the set

$$C_i(V) = \{c_i^A \circ v \mid v \in V\}$$

is called the i-th component of V, $(1 \leq i \leq n)$.

Evidently, if $V \subseteq (A_1 \times A_2 \times \ldots \times A_n)^U$ is a G-object, then $C_i(V) \subseteq A_i^U$. Hence, $C_i(V)$ is itself a G-object. Now, $C_i(V)$ may or may not be a unary G-object. For example, if $V \subseteq (A_1 \times (A_{21} \times A_{22}) \times A_3)^U$, then $C_2(V)$ is 2-ary.[4] Finally, in the given case, we note $(c_i^A \circ v): U \to A_i$ and for all $t \in U$,

$$v(t) = ((c_1^A \circ v)(t), (c_2^A \circ v)(t), \ldots, (c_n^A \circ v)(t))$$

Hence, the concept of the components of a G-time function is no different than the concept of the components of any other function.

A Duality

As may already be apparent to the reader, there exists a basic duality between the concepts of an n-ary G-system and an n-ary G-object via the notion of components of functions. This is a useful duality for some purposes. We indicate the duality by defining:

If $V \subseteq A^U$ is an n-ary G-object, then the dual of V is the set

$$S = \{(c_1^A \circ v, c_2^A \circ v, \ldots, c_n^A \circ v) \mid v \in V\}$$

Clearly, if $V \subseteq (A_1 \times A_2 \times \ldots \times A_n)^U$, then $S \subseteq A_1^U \times A_2^U \times \ldots \times A_n^U$. Thus, S is an n-ary G-system. Also, for all i $(1 \leq i \leq n)$,

$$P_i S = C_i(V)$$

Now then, evidently, every n-ary G-object thus has a (unique) dual which

---

[4] The unambiguity of the dimension of a G-object thus comes down to the fact that Cartesian product of sets is not an associative operation. See Suppes, op. cit.

is an n-ary G-system. The converse is also true. That is, if S is an n-ary G-system, then there exists a unique n-ary G-object V such that S = Dual(V). In fact, let $S \subseteq A_1^U \times A_2^U \times \dots \times A_n^U$ be a G-system. Then, if $s = (s_1, s_2, \dots, s_n) \in S$, associate the map $v_s : U \to (A_1 \times A_2 \times \dots \times A_n)$ such that

$$v_s(t) = (s_1(t), s_2(t), \dots, s_n(t)), \qquad (t \in U)$$

and let

$$V = \{ v_s \mid s \in S \}$$

Then, S = Dual(V). Thus, there exists a 1:1 correspondence between the class of n-ary G-objects and the class of n-ary G-systems associated with a given space relation $A \subseteq A_1 \times A_2 \times \dots \times A_n$.

Input-Output Systems

With the above preliminaries out of the way, we can begin the axiomatic development of our formal concept of a system so as to make it appear more and more like the mathematical models of real systems which we deal with in engineering. One of the most important problems in general systems theory is the investigation and delineation of the principle of cause-and-effect as a premise underlying the behavior of real systems. Our formalized concept of a general time system, the n-ary G-system, is not revealing as it stands of cause-effect (input-output; stimulus-response) behavior except by the most liberal interpretation. This is deliberately the case. By starting in an extremely general set-up, we allow ourselves the freedom of an axiomatic introduction of formal properties to render our systems more and more interpretable as cause-effect (or as Zadeh [5] puts it, oriented) entities. In the process, we can theorize about the "universality" of this principle.

The first step in our delineation of the cause-and-effect idea

in systems theory is to specialize the n-ary G-system to the "input-output system" and establish a second duality concept; namely, universal representation of general time systems as input-output systems. This is a matter of introducing the concepts of "inputs" and "outputs". The basic methodology we employ is due to Mesarović [6].

If G is an ordered group as above, a G-__input-output__ __system__ is any 2-ary G-system, i.e. $S \subseteq A^U \times B^U$. In this case, the G-objects A^U and B^U are called, respectively, a __cause__ __object__ and an __effect__ __object__ for S. The set A is an __input__ __sp__ and B is an __output__ __space__ for S. The sets

$$\mathcal{D}S = \{x \mid (\exists y): xSy\}$$
$$\mathcal{R}S = \{y \mid (\exists x): xSy\}$$

(which are unique for S) are called the __input__ __set__ and the __output__ __set__ of S. In general, if S is a G-input-output system, then

$$\mathcal{D}S = \mathcal{P}_1 S$$
$$\mathcal{R}S = \mathcal{P}_2 S$$

and
$$S \subseteq \mathcal{D}S \times \mathcal{R}S$$

i.e. an input-output system is a relation on its input set and its output set. Thus, we satisfy one of the basic positions (due to Mesarović) which we stated at the outset.

We next formalize the concept of a multi-variable input-output system which is perhaps the most fundamental of the block diagram concepts: If S is a G-input-output system, then S is __single-variable__ if and only if both $\mathcal{D}S$ and $\mathcal{R}S$ are unary G-objects. If S is not single-variable, it is __multi-variable__. Thus, a multi-variable G-input-output system is a relation of the form

$$S \subseteq (A_1 \times A_2 \times \ldots \times A_n)^U \times (B_1 \times B_2 \times \ldots \times B_m)^U$$

where either $n > 1$ or $m > 1$ or both. If

$S \subseteq (A_1 \times A_2 \times \ldots \times A_n)^U \times (B_1 \times B_2 \times \ldots \times B_m)^U$ is a (possibly single-variable) G-system, then the set $C_i(\mathcal{E}S)$ is the i-th input port of S ($1 \leq i \leq n$) and the set $C_j(\mathcal{R}S)$ is the j-th output port of S ($1 \leq j \leq m$). S is isolated if and only if

$$\text{Dual}(\mathcal{E}S) = C_1(\mathcal{E}S) \times C_2(\mathcal{E}S) \times \ldots \times C_n(\mathcal{E}S)$$

S is non-cohesive if and only if

$$\text{Dual}(\mathcal{R}S) = C_1(\mathcal{R}S) \times C_2(\mathcal{R}S) \times \ldots \times C_m(\mathcal{R}S)$$

It is immediate from the definition that every single-variable G-system is isolated and non-cohesive. Moreover, we see a multi-variable G-system S is isolated [non-cohesive] if and only if $\text{Dual}(\mathcal{E}S)$ [$\text{Dual}(\mathcal{R}S)$] is an improper G-system.

Our interpretation of the property of isolation is, of course, that all of the input ports of the system are "independent" of each other. Similarly, non-cohesion is the property that all of the output ports are "independent" of each other. Cohesion and a number of other concepts related to the concept of interaction are discussed at length in the companion paper by L. Birta.

We consider next the question of whether or not in some definite sense the input-output systems are "rich" in the class of general time systems. We formalize the issue with the following definition:

Let $S \subseteq A^U \times B^U$ be a G-input-output system with $A \subseteq A_1 \times A_2 \times \ldots \times A_n$ and $B \subseteq B_1 \times B_2 \times \ldots \times B_m$. If S' is an (n+n)-ary G-system, then S is an input-output representation for S' if and only if

$$xSy \Longleftrightarrow (c_1^A \circ x, c_2^A \circ x, \ldots, c_n^A \circ x, c_1^B \circ y, c_2^B \circ y, \ldots, c_m^B \circ y) \in S'$$

Now, it is a conceptually important result that in the given sense every n-ary G-system ($n \geq 2$) has an input-output representation. The proof of this fact is given by Mesarovic in [6]. The issue is simply choosing

a cause object and an effect object for the representation system. That
is, if $S \subseteq A_1^U \times A_2^U \times \ldots \times A_n^U$, we choose some integer m $(1 \leq m < n)$
and set $A = A_1 \times A_2 \times \ldots \times A_m$ and $B = A_{m+1} \times A_{m+2} \times \ldots \times A_n$. Then,
the relation $S' \subseteq A^U \times B^U$ such that

$$x S' y \iff (c_1^A \circ x, c_2^A \circ x, \ldots, c_m^A \circ x, c_1^B \circ y, c_2^B \circ y, \ldots, c_{n-m}^B \circ y) \in S$$

is an input-output representation for S. Thus, in the sense of

representability, input-output systems are, in fact, "rich" in the

class of general time systems. For this reason, we can justify

restricting further attention to input-output systems. That is, if we

do not disallow the multi-variable case, then our finding in an

important sense shall be valid for the class of all n-ary G-systems.

Operations On Systems

There are a number of different ways in which systems can be

combined to yield other systems. In the case of input-output systems,

these arise in two distinct ways, (i) through interconnections of

systems such as the series, parallel, and feedback interconnections,

and (ii) through formalizing the evolution of a given system in time.

Interconnections of systems, of course, are primary concepts in the

block diagram description of systems. The latter concept is not. In

this section, we shall formalize the series interconnection of systems

and the concept of "sectioning" a given system (which is (ii) above).

The parallel and feedback interconnections can also be formalized for

G-input-output systems, but play no role in our development of the

concept of state; hence, they are not needed.

Let us first address ourselves to the concept of the series

interconnection of input-output systems: Henceforth, let A and B

be fixed sets and (as before) let $G = (T, +, U, <)$ be a given ordered

group. Then, define Δ to be the class of all G-input-output systems

with input space A and output space B, i.e. let

$$\mathcal{S} = \{ S \mid S \subseteq A^U \times B^U \}$$

Then, if S and S' are arbitrary G-input-output systems, then the
series interconnection of S and S' is simply the composition relation
(S' o S). Thus, if $S \subseteq A^U \times B^U$ and $S' \subseteq C^U \times D^U$, then
$(S' \ o \ S) \subseteq A^U \times D^U$. In the particular case that A = C and B = D,
$(S' \ o \ S) \subseteq A^U \times B^U$. Therefore, for all $S, S' \in \mathcal{S}$, $(S' \ o \ S) \in \mathcal{S}$.
Thus, the class $\mathcal{S}$ of input-output systems is closed under series
interconnection.

Now, actually, for proper interpretation in the engineering context,
we must qualify the definition of series interconnection slightly.
This is because engineering systems must generally satisfy an inter-
connectability condition; namely, $\mathcal{R}S \subseteq \mathcal{D}S'$. Thus, we formalize:
If S and S' are G-input-output systems, then the series interconnection
(S' o S) is proper if and only if $\mathcal{R}S \subseteq \mathcal{D}S'$. Clearly, whenever (S' o S)
is a proper series interconnection, then (as we noted above)
$\mathcal{E}(S' \ o \ S) = \mathcal{D}S$.

The concept of "sectioning" a given input-output system is some-
what more difficult to formalize than the notion of the series inter-
connection of systems: If $S \in \mathcal{S}$ and $t \in U$, then the (normalized) t-section
of S is the relation

$$S_t = \{ (x_t \ o \ e_t, y_t \ o \ e_t) \mid x S y \}$$

where

$$x_t = \{ (t', x(t')) \mid t < t' \}$$
$$y_t = \{ (t', y(t')) \mid t < t' \}$$

and $e_t : T \to T$ is the t-left translation on T. Also, if $S \in \mathcal{S}$, define
$S_0 = S$ where 0 is the identity of T.

Now, $\mathcal{D} x_t = \{t' \mid t < t'\}$. Therefore, since $x: U \to A$ and $c_t: T \to T$ and c_t is monotonic, it follows that $(x_t \circ c_t): U \to A$. In fact,

$$t' \in \mathcal{D}(x_t \circ c_t) \Leftrightarrow t' \in T \ \& \ c_t(t') \in \mathcal{D} x_t \Leftrightarrow t' \in T \ \& \ t < c_t(t')$$

$$\Leftrightarrow t' \in T \ \& \ t < t + t' \Leftrightarrow t' \in T \ \& \ (t + 0) < (t + t')$$

$$\Leftrightarrow t' \in T \ \& \ 0 < t' \Leftrightarrow t' \in U$$

i.e. $\mathcal{D}(x_t \circ c_t) = U$. Similarly, we see $(y_t \circ c_t): U \to B$. From this, it follows that

$$S_t \subseteq A^U \times B^U$$

Finally, then, for all $S \in \mathcal{S}$ and all $t \in U$, $S_t \in \mathcal{S}$ (!) In words, S_t is that part of the system defined for time greater than t shifted backwards in time to the starting time 0. Under interpretation, S_t is what the system S "looks like" starting at time t.

Now, from the fact that $S_t \in \mathcal{S}$, there evidently exists a function $\pi: \mathcal{S} \times \overline{U} \to \mathcal{S}$ (where $\overline{U} = U \cup 0$) such that

$$\pi(S,t) = S_t$$

We shall call π the <u>motion</u> in $\mathcal{S}$. Now, very importantly, it can be proved that π satisfies the following properties; namely,

    (i) $\pi(S,0) = S$, $(S \in \mathcal{S})$

    (ii) $\pi(S, t + t') = \pi(\pi(S,t),t')$, $(S \in \mathcal{S} \ \& \ t, t' \in \overline{U})$

(The condition (ii) is of course equivalent to $S_{t+t'} = (S_t)_{t'}$.)

Thus, the motion π is always available for characterizing the "evolution" of an input-output system through time. It can be seen that this notion is important conceptually in engineering.

The operations of series interconnection and sectioning are not unrelated. That is, if S and $S' \in \mathcal{S}$, then for all $t \in U$,

$$(S' \circ S)_t = S'_t \circ S_t$$

In fact,

$$x(S' \circ S)_t y \Leftrightarrow (\exists u)(\exists v): u(S' \circ S)v \;\&\; x = u_t \circ c_t \;\&\; y = v_t \circ c_t$$
$$\Leftrightarrow (\exists u)(\exists v)(\exists w): uSw \;\&\; wS'v \;\&\; x = u_t \circ c_t \;\&\; y = v_t \circ c_t$$
$$\Leftrightarrow (\exists w): xS_t(w_t \circ c_t) \;\&\; (w_t \circ c_t)S'_t y$$
$$\Leftrightarrow x(S'_t \circ S_t)y$$

Thus, the t-section of the series interconnection of two systems in
is the series interconnection of the t-sections of the systems.

Finally, then, on the class of G-input-output systems $\mathcal{S}$, we
have two basic operations; namely, (i) series interconnection
$\circ: \mathcal{S} \times \mathcal{S} \to \mathcal{S}$ (which is a semi-group operation) and (ii) sectioning,
i.e. the motion $\tau: \mathcal{S} \times \overline{U} \to \mathcal{S}$ (with $(\overline{U},+)$ being a semi-group with
identity). Finally, these two operations are connected by the
conditions

(i) $\tau(S,0) = S$

(ii) $\tau(S, t + t') = \tau(\tau(S,t), t')$

(iii) $\tau(S' \circ S, t) = \tau(S', t) \circ \tau(S, t)$[5]

Non-Anticipatory Functions

Perhaps the most fundamental property of input-output systems
is the property of non-anticipation. Roughly speaking, an input-output
system is non-anticipatory if the "present value" of any output of the
system does <u>not</u> depend on any "future values" of the corresponding
system input. This property is, of course, an intuitive one well-known
in engineering. The importance of the concept lies in the fact that
non-anticipation is equatible with (or at least a necessary condition
for) the "physical realizability" of the system. Of course, a better
way to say the same thing is: Non-anticipation is a property which is
almost universally valid for the class of mathematical models employed

-----

[5] Thus, the set-up $(\mathcal{S}, \tau, \circ)$ is a (quasi-) module algebraically.

to describe real physical systems.

Now, in attempting to formalize the intuitive concept (above) of a non-anticipatory system, we encounter an immediate problem. The above statement of non-anticipation implies the system is a function; yet, this does not seem to be a necessary condition. Hence, a non-trivial task is to formalize the concept of a non-anticipatory system in the case of input-output systems which are not functional. We shall here solve the problem by carrying out the functional case completely; then, in the light of the results, we shall propose a definition in the non-functional case.[6]

Proceeding with our axiomatic development, then, we define: If $S \in \mathcal{S}$, then S is a non-anticipatory function if and only if (i) S is a function, i.e. $S: \mathcal{D}S \to \mathcal{R}S$ and (ii) for all $x,x' \in \mathcal{D}S$ and all $t \in U$,

$$x_0^t = x'_0^t \implies S(x)(t) = S(x')(t)$$

where

$$x_0^t = \{(t', x(t')) \mid 0 < t' \leq t\}$$
$$x'_0^t = \{(t', x'(t')) \mid 0 < t' \leq t\}$$

Regarding this definition, we realize that condition (ii) could be replaced by the equivalent condition (ii') for all $x,x' \in \mathcal{D}S$ and all $t \in U$,

$$x_0^t = x'_0^t \implies (S(x))_0^t = (S(x'))_0^t$$

As we have previously stated, it turns out that non-anticipation is the axiomatic cornerstone on which the concept of state in systems theory can be founded. We begin to make clear this fact when we examine the closure properties of the operations on systems previously introduced

in the special case they are applied to non-anticipatory functions.

Henceforth, let $\mathcal{N}$ be the class of all G-input-output systems with input space A and output space B which are non-anticipatory functions, i.e.

$$\mathcal{N} = \{ S \mid S \subseteq A^U \times B^U \ \& \ S \text{ is a non-anticipatory function} \}$$

Then, $\mathcal{N} \subseteq \mathcal{S}$.

We note two things: First, if S and S' are non-anticipatory functions, then their series interconnection (S' o S) is a non-anticipatory function. In fact, since S and S' are functions, (S' o S) is a function. Moreover, for all $x, x' \in \mathcal{D}(S' \text{ o } S)$ and all $t \in U$,

$$x_0^t = x'^t_0 \Rightarrow (S(x))_0^t = (S(x'))_0^t \Rightarrow S'(S(x))(t) = S'(S(x'))(t)$$
$$\Rightarrow (S' \text{ o } S)(x)(t) = (S' \text{ o } S)(x')(t)$$

Thus, (S' o S) is non-anticipatory. In particular, if $S, S' \in \mathcal{N}$, then $(S' \text{ o } S) \in \mathcal{N}$ and, hence, o: $\mathcal{N} \times \mathcal{N} \to \mathcal{N}$. Second, if $S \in \mathcal{N}$ and $t \in U$, then S_t need not be a non-anticipatory function. In fact, we readily see that S_t is not even a function in general. Thus, we find that the sections of a non-anticipatory function are in general anticipatory! At first, this is perplexing but a bit of further investigation unravels the mystery. We define:

If S is a non-anticipatory function, if $x \in \mathcal{D}S$, and $t \in U$, then the (x,t)-appearance of S is the relation

$$S_t^x = \{ (x'_t \text{ o } e_t, y_t \text{ o } e_t) \mid x'Sy \ \& \ x'^t_0 = x_0^t \}$$

Also, for all $x \in \mathcal{D}S$, define $S_0^x = S$. Then, using the same arguments as we employed before in introducing the sections of a system, we find

$$S_t^x \subseteq A^U \times B^U$$

Hence, if $S \in \mathcal{N}$, $x \in \mathcal{B} S$, and $t \in U$, then $S_t^x \in \mathcal{J}$. More importantly, it can be proved that S_t^x is a non-anticipatory function. In other words, $S_t^x \in \mathcal{N}$. Finally, we note

$$S_t = \bigcup_{x \in \mathcal{B} S} S_t^x$$

and, hence, we find that if S is a non-anticipatory function and $t \in U$, then S_t has a natural decomposition into non-anticipatory functions; namely, the set

$$\mathcal{N}_S(t) = \left\{ S_t^x \mid x \in \mathcal{B} S \right\}$$

Now, from the fact that $S_t^x \in \mathcal{N}$, it follows that we can construct another function (similar to the notion of S) which describes the evolution of a non-anticipatory function in terms of its non-anticipatory appearances. First, we define the relation $D \subseteq \mathcal{N} \times A^U \times U$ such that

$$(S, x, t) \in D \Longleftrightarrow x \in \mathcal{B} S$$

Then, there evidently exists a function $\mathcal{T} : D \to \mathcal{N}$ such that

$$\mathcal{T}(S, x, t) = S_t^x$$

$\mathcal{T}$ we shall call the __branching__ .. in $\mathcal{N}$. Very importantly, it can be proved that $\mathcal{T}$ satisfies the following properties whenever the appropriate images of $\mathcal{T}$ are defined:

(i) $\mathcal{T}(S, x, 0) = S$

(ii) $x_0^t = x'_0^t \Longrightarrow \mathcal{T}(S, x, t) = \mathcal{T}(S, x', t)$

(iii) $\mathcal{T}(S, x, t + t') = \mathcal{T}(\mathcal{T}(S, x, t), x_t \circ e_t, t')$

(The condition (iii) is of course equivalent to $S_{t+t'}^x = (S_t^x)_{t'}^{x_t} \circ e_t$.)

Thus, for a non-anticipatory function S, the branching $\mathcal{T}$ is always available for tracing the evolution of S in time in terms of non-anticipatory functions.

The Concept of State

Having examined the non-anticipatory functions, we wish to set down a definition for a G-input-output system which is non-anticipatory but not a function. In the light of the foregoing, it is natural to define:

If $S \in \Delta$, then S is a non-anticipatory system if and only if there exists some non-anticipatory function $S' \in \mathcal{H}$ and some $t \in U$ such that

$$S = S'_t$$

In words, an input-output system is non-anticipatory if it is a section of a non-anticipatory function. Under this definition, we note every non-anticipatory function is a non-anticipatory system, i.e. if $S \in \mathcal{H}$, then (trivially)

$$S = S_0$$

Now, very importantly, we discover that under this definition non-anticipatory systems have states and all of the usual machinery of state transitions in systems theory [5]:

Let $S \in \Delta$ be a non-anticipatory system, i.e. assume $S = S'_t$ where $S' \in \mathcal{H}$. Then, the set

$$\mathcal{H}_{S'}(t) = \left\{ S'^x_t \mid x \in \mathcal{D}S' \right\}$$

is a set of initial states for S. Clearly, $\mathcal{H}_{S'}(t) \subseteq \mathcal{H}$ and

$$S = \cup \mathcal{H}_{S'}(t)$$

and, from the latter, it follows that for all $x \in A^U$ and all $y \in B^U$,

$$xSy \iff (\exists f): y = f(x), \qquad (f \in \mathcal{H}_{S'}(t))$$

The set

$$\mathcal{H}^t_{S'} = \left\{ S'^x_t \mid x \in \mathcal{D}S' \ \& \ t \leq t' \right\}$$

is a set of states for S. In general, $\mathcal{H}_{S'}(t) \subseteq \mathcal{H}^t_{S'}$ and, hence, every initial state of S is a state of S.

From these definitions, we see (i) every state of a non-anticipatory system is a non-anticipatory function (ii) a non-anticipatory function is a non-anticipatory system with precisely one initial state and (iii) a non-anticipatory system is an input-output system which at some earlier time had precisely one initial state.

If $S \in \mathcal{S}$ is non-anticipatory (i.e. $S = S'_t$ with $S' \in \mathcal{N}$) and $t' \in U$, then $S_{t'} \in \mathcal{S}$ is a non-anticipatory system, i.e. every section of a non-anticipatory system is a non-anticipatory system. In fact,

$$S_{t'} = (S'_t)_{t'} = S'_{t+t'}$$

Moreover, then, the set

$$\mathcal{N}_{S'}(t + t') = \{ S^x_{t+t'} \mid x \in \mathcal{D} S' \}$$

is a set of initial states for $S_{t'}$. Also, $\mathcal{N}_{S'}(t + t') \subseteq \mathcal{N}^t_{S'}$ and, in fact,

$$\mathcal{N}^t_{S'} = \bigcup_{t' \in \bar{U}} \mathcal{N}_{S'}(t + t')$$

Now, using the property that $S^x_{t+t'} = (S^x_t)^{x_t \circ e_t}_{t'}$, we see that

$$\mathcal{N}_{S'}(t + t') = \{ f^x_{t'} \mid f \in \mathcal{N}_{S'}(t) \ \& \ x \in \mathcal{D} f \}$$

Thus, the set $\mathcal{N}_{S'}(t + t')$ of initial states for $S_{t'}$ can be "computed" directly from the set $\mathcal{N}_{S'}(t)$ of initial states for S.

If S is a non-anticipatory system as above, then the branching τ in $\mathcal{N}$ is called a _state transition function_ for S. As we noted before, τ has the properties,

     (i) $\tau(S,x,0) = S$

     (ii) $x^t_0 = x'^t_0 \implies \tau(S,x,t) = \tau(S,x',t)$

     (iii) $\tau(S,x,t + t') = \tau(\tau(S,x,t), x_t \circ e_t, t')$

Thus, by (ii), the state transitions of S are made in a non-anticipatory fashion. Also, by (iii), they obey a semi-group property.

Finally, every non-anticipatory system has an <u>output function</u> [5];
namely, the function $\sigma: D \to B$ such that

$$\sigma(S,x,t) = S(x)(t)$$

That is, in general, it can be proved: If $(S,x,t + t') \in D$, then
$(S,x,t) \in D$ and $(S_t^x, x_t \circ e_t, t') \in D$ and

$$S(x)(t + t') = S_t^x(x_t \circ e_t)(t')$$

Therefore, the function σ has the property

$$\sigma(S,x,t + t') = \sigma(\tau(S,x,t), x_t \circ e_t, t')$$

for all $(S,x,t + t') \in D$.

Summarizing, then, if $S \in \mathcal{A}$ is a non-anticipatory system, then
there exists a subset $\eta' \subseteq \eta$, a relation $D \subseteq \eta \times A^U \times \tilde{U}$, and
two maps $\tau: D \to \eta$ and $\sigma: D \to B$ such that

(i) $x S y \iff (\exists f)(\forall t): y(t) = \sigma(f,x,t),$ $(f \in \eta')$

(ii) $\sigma(f,x,t + t') = \sigma(\tau(f,x,t), x_t \circ e_t, t'),$ $(f \in \eta)$

(iii) $\tau(f,x,t + t') = \tau(\tau(f,x,t), x_t \circ e_t, t'),$ $(f \in \eta)$

(iv) $x_0^t = x'_0^t \implies \sigma(f,x,t) = \sigma(f,x',t),$ $(f \in \eta)$

(v) $x_0^t = x'_0^t \implies \tau(f,x,t) = \tau(f,x',t),$ $(f \in \eta)$

Thus, every non-anticipatory system has a set of initial states, a set
of states, a state transition function, and an output function.

Acknowledgment

References

1   Windeknecht, T. G., Mathematical systems theory: causality, _J. Math. Sy. Th._ 1, 1967 (to appear).

2   Suppes, P., "Axiomatic Set Theory," Van Nostrand, Princeton, New Jersey, 1960.

3   Jacobson, N.,"Lectures in Abstract Algebra," Vol.3 (Theory of Fields and Galois Theory), Van Nostrand, Princeton, New Jersey, 1964.

4   Herstein, I. N., "Topics In Algebra," Blaidell, New York, 1964.

5   Zadeh, L. A., The concept of state in system theory _in_ "Views On General Systems Theory," Wiley, New York, 1964.

6   Mesarovic, M. D., Foundations for a general systems theory _in_ "Views On General Systems Theory," Wiley, New York, 1964.

7   Windeknecht, T. G., Mathematical systems theory: determinism, _Conference On Machines, Languages, and Semi-Groups_, Pacific Grove, California, 1966.

8   Windeknecht, T. G., Concerning an algebraic theory of systems, _Conference On Computer Science and Systems_, London, Canada, 1965. Univ. of Toronto Press, 1967 (to appear).

9   Windeknecht, T. G. and M. D. Mesarovic, On general dynamical systems and finite stability _in_ "Differential Equations and Dynamical Systems," Academic Press, New York, 1967.